Dynamic matrix factorization-based collaborative filtering in movie recommendation services

Vuong Luong Nguyen[1*], Trinh Quoc Vo[1], Hoai Thi Thuy Nguyen[1]

[1]FPT University, Danang, Vietnam
[*]Corresponding author: vuongnl3@fe.edu.vn

ARTICLE INFO

DOI:10.46223/HCMCOUJS.tech.en.14.1.2922.2024

Received: August 21st, 2023

Revised: September 27th, 2023

Accepted: October 23rd, 2023

Keywords:

collaborative filtering; matrix factorization; recommendation system

ABSTRACT

Movies are a primary source of entertainment, but finding specific content can be challenging given the exponentially increasing number of movies produced each year. Recommendation systems are extremely useful for solving this problem. While various approaches exist, Collaborative Filtering (CF) is the most straightforward. CF leverages user input and historical preferences to determine user similarity and suggest movies. Matrix Factorization (MF) is one of the most popular Collaborative Filtering (CF) techniques. It maps users and items into a joint latent space, using a vector of latent features to represent each user or item. However, traditional MF techniques are static, while user cognition and product variety are constantly evolving. As a result, traditional MF approaches struggle to accommodate the dynamic nature of user-item interactions. To address this challenge, we propose a Dynamic Matrix Factorization CF model for movie recommendation systems (DMF-CF) that considers the dynamic changes in user interactions. To validate our approach, we conducted evaluations using the standard MovieLens dataset and compared it to state-of-the-art models. Our preliminary findings highlight the substantial benefits of DMF-CF, which outperforms recent models on the MovieLens-100K and MovieLens-1M datasets in terms of Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) metrics.

1. Introduction

In the contemporary landscape, users are confronted with an overwhelming array of choices. E-commerce retailers and providers such as Netflix present an extensive range of options, necessitating users to make selections that resonate with their interests. The pivotal step of aligning users with the most fitting products is key to enhancing user contentment and engendering loyalty. In light of this, Recommender Systems (RSs) (Adomavicius & Tuzhilin, 2005; Nguyen, Hong, Jung, & Sohn, 2020; Nguyen & Jung, 2020; Nguyen, Nguyen, & Jung, 2020) have surged in popularity within e-commerce platforms. These systems analyze user interest patterns to offer personalized recommendations tailored to individual tastes. Consequently, e-commerce giants like Amazon, Netflix, Flipkart, and YouTube have seamlessly integrated RSs into their online platforms, significantly elevating the overall user experience. The strategies employed by RSs can be broadly categorized into two strategies that are Content-Based (CB) filtering (Nguyen, Nguyen, et al., 2020) and CF (Nguyen, Hong, et al., 2020; Nguyen, Vo, & Nguyen, 2023). The CB approach entails crafting user-profiles and product catalogs. A user profile encompasses demographic details, while product descriptions - for instance, those of movies - might encompass genre, box office popularity, cast, and more. However, CB strategies often necessitate the involvement of experts to curate external information that might not be readily available. In contrast, the CF methodology operates on the user's historical behavior, eschewing the explicit creation of profiles. CF examines user-item relationships and interdependencies among items to unearth novel correlations.

The inherent strength of the CF approach lies in its domain-independent nature, rendering CF techniques more accurate compared to CB. Nevertheless, CF grapples with the challenges posed by new users and products, the cold start, and the data sparsity. The core technique of CF is built upon neighborhood and model-based approaches. Regarding the neighborhood approaches, these techniques try to identify the best similar users or items through similarity scores (Nguyen, Nguyen, Jung, & Camacho, 2023; Nguyen, Nguyen, & Jung, 2021; Nguyen, Vo, et al., 2023). The exemplar of this approach is user-based and item-based CF. Latent Factor (LF) (Koren, Bell, & Volinsky, 2009) techniques present an alternative to neighbor methods. LF models decode user-item interactions into latent or concealed factors linked to users and items. For instance, MFs for movies gauge dimensions like genre and suitability for children, while user factors gauge a user's affinity for high-scoring items concerning item factors. The widely recognized Matrix Factorization (MF) (Koren et al., 2009) technique falls under the LF umbrella, breaking interaction matrices into user and item factors.

Motivated by the limitations of traditional MF, particularly its lack of consideration for the temporal dynamics of user-item interactions, we construct an LF model capable of accommodating dynamic aspects of user preferences. We incorporate dynamic features into the traditional MF framework to achieve this and then validate the efficacy of the DMF-CF model on MovieLens datasets. The salient contributions of this study are summarized as follows:

- Introduction of a CF-based movie recommender system enriched with dynamic features within the MF paradigm to effectively address the dynamic nature of user preferences.

- Execution of comprehensive experiments to substantiate the superiority of the proposed system over baseline methods that use the Movielens 100K and 1M datasets.

The remainder of this paper is organized as follows. The next section will briefly overview a few relevant, recent research on recommendation systems based on MF. The problem formulation and proposed method are details described in Section 3. Section 4 provides our experiments consisting of settings, evaluation metrics, and experiential results. Finally, we conclude and show the directions for future work in Section 5.

2. Related work

Aciar, Aciar, Collazos, and Gonzalez (2016) presented an innovative user recommender system that harnessed knowledge, availability, and reputation derived from interactions within online forums. This study showcased the potential of incorporating user-generated content and social interactions to enhance recommendation accuracy. Moving towards algorithmic advancements, Anwar, Uma, and Srivastava (2021) introduced *Rec-CFSVD++*, a recommendation system that skillfully merged CF with Singular Value Decomposition (SVD)++. This integration not only harnessed the power of MF modeling but also incorporated user and item biases, thereby contributing to more nuanced and accurate recommendations. Such hybrid methodologies have gained traction due to their ability to capture intricate user preferences.

Time-sensitive recommendations garnered significant attention as well. Ding and Li (2005) introduced the concept of "Time Weight CF," where time-based weights were integrated into recommendation models. This approach recognized that user preferences evolve over time and leveraged temporal patterns to enhance the quality of recommendations. He, Zhang, Kan, and Chua (2016) ventured into the domain of online recommendations with implicit feedback. Their *Fast MF* methodology addressed the challenge of efficiently processing and updating recommendations as new interactions flowed in. This was particularly crucial in scenarios with rapid user engagement and constantly evolving preferences.

Temporal dynamics emerged as a focal point for several researchers. Karahodza, Supic, and Donko (2014) proposed a time-aware recommender system, emphasizing changes in group users' preferences over time. Such temporal adaptability is crucial in capturing shifting user interests and catering to dynamic recommendation scenarios. Lathia, Hailes, and Capra (2009) further explored temporal CF with adaptive neighborhoods, acknowledging that user preferences and item relevance may vary based on the timing of interactions. Lee, Park, and Park (2008, 2009) empirically studied the effectiveness of temporal information in recommendations, highlighting the potential of incorporating temporal features as implicit ratings to enhance the accuracy of predictions. Koren's contributions (Koren, 2008, 2009) significantly shaped the field by introducing multifaceted CF models. These approaches combined MF with neighborhood-based algorithms, effectively capturing both MFs and local item-item relationships. Koren (2008) underscored the importance of factorization combined with neighborhood-based models in addressing the limitations of each approach when used in isolation. Koren's exploration into CF with temporal dynamics (Koren, 2009) further enriched the field by integrating the temporal dimension into recommendation models, enabling the system to adapt to evolving user preferences.

In the context of evolving content consumption patterns, Liu, Zhao, Xiang, and Yang (2010) introduced *Online Evolutionary CF*, a novel approach that catered to the dynamic nature of online platforms. By incorporating evolutionary techniques, this study addressed the challenge of maintaining recommendation quality in the face of rapidly changing user behaviors. Meng, Zheng, Tao, and Liu (2016) delved into the domain of user-specific rating prediction for mobile applications. Their weight-based MF technique highlighted the significance of tailoring recommendations to individual users' preferences, catering to the diverse needs of mobile app users. Furthermore, Ricci and Nguyen (2007) explored the acquisition and revision of preferences in critique-based mobile recommender systems. This approach showcased the potential of integrating user feedback and critiques into the recommendation process, contributing to more user-centric and context-aware recommendations. Takács, Pilászy, Németh, and Tikk (2008) made remarkable strides by contributing to the Netflix Prize Problem, utilizing MF and neighbor-based algorithms to enhance movie recommendations. This research showcased the practical implications of MF in real-world, large-scale recommendation scenarios. Interested readers can find recent studies related to recommendation systems in several survey papers (Raza & Ding, 2022; Roy & Dutta, 2022; Wang, Ma, Zhang, Liu, & Ma, 2023).

3. Proposed model

3.1. Problem formulation

MF-based CF is a powerful technique for building recommendation systems that predict user preferences for items based on their interactions with a historical dataset. However, traditional MF-based CF methods often overlook the temporal dynamics and evolving preferences of users. To address this limitation, we propose a Dynamic MF-based Collaborative Filtering (DMF-CF) approach that incorporates time-varying features to enhance recommendation accuracy.

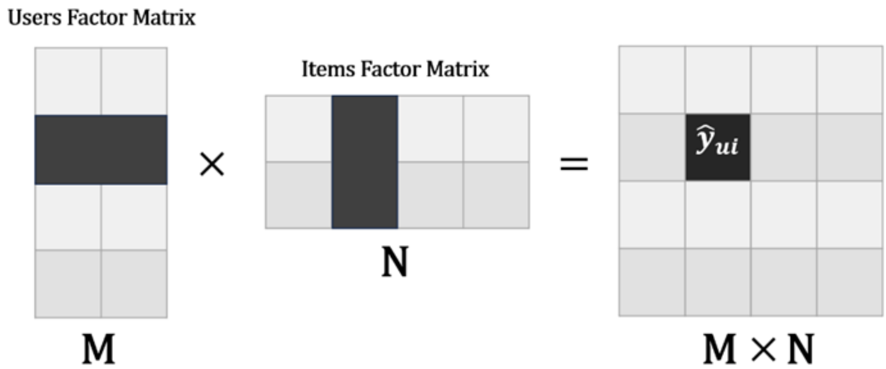

Figure 1. Illustration of MF model regarding M users factor matrix and N items factor matrix

In particular, the input are i) user-item interaction matrix R of dimensions $m \times n$, where m is the number of users and n is the number of items; ii) timestamps associated with user-item interactions, capturing when the interactions occurred (temporal information); and iii) additional time-varying features that provide context for user preferences, such as trending items, or contextual changes (dynamic features). The output is user-item MF matrices U and V of dimensions $m \times k$ and $n \times k$, respectively, where k is the number of latent factors.

3.2. Matrix factorization with dynamic features

The goal of DMF-CF is to factorize the user-item interaction matrix R into user and item MF matrices U and V, while leveraging temporal information and dynamic features to capture evolving user preferences and item characteristics. The illustration of the MF model is shown in Figure 1. This allows us to generate accurate recommendations that adapt to changing user behavior over time. The main purpose of the latent factor model is to factorize the matrix into two matrices (low rank) as formulated in Equation 1.

$$y_{ui} = U \cdot V^T \tag{1}$$

Generally, several methods, such as SVD and PCA, are applied to the rank rating matrix. However, SVD can not process with the high portion of missing ratings. Therefore, a solution to face this challenge is a regularized SVD model. The regularized SVD is formulated as follows:

$$\hat{y}_{ui} = U^T \cdot V + \omega_u + \omega_i \tag{2}$$

In Equation 2, the ω_u is the user bias while the ω_i denotes the item bias. U and V are the user and item factors, respectively. The regularized SVD model was trained by optimizing Equation 2; this process is formulated as follows:

$$min_{U,V} \sum (y_{ui} - u^T v - \omega_u - \omega_i)^2 + \lambda(\omega_u^2 + \omega_i^2 + \|u\|^2 + \|v\|^2) \tag{3}$$

Following the above idea, we added the dynamic change parameter of item population and user preferences. The Equation 2 is rewritten as follows:

$$\hat{y}_{ui}^t = U_i^T \cdot V_j^T + \omega_u(t) + \omega_i(t) + \delta(U_i \times t_0) + U_i(t) \tag{4}$$

Where the dynamic features are $U_i(t)$, $\omega_u(t)$, and $\omega_i(t)$ that are used to generate the prediction $\hat{y}_{ui}^t$ dynamically at time t. The $\delta(U_i \times t_0)$ represents the time function that changes rely on the users. The Dynamic MF-CF (DMF-CF) aims to optimize the following objective function:

$$\mathcal{L} = min_{U,V} \sum_{(i,j) \in I} \left(r_{ij} - u_i^T v_j\right)^2 + \lambda \left(\omega_u(t) + \omega_i(t) + \|u_i\|^2 + \left\|v_j\right\|^2\right) \tag{5}$$

Where u_i represents the latent factors for user i, v_i represents the latent factors for item j, r_{ij} is the actual rating given by user i to item j, and λ is the regularization parameter. Additionally, the model integrates temporal information and dynamic features by incorporating them into the MF calculations:

$$u_i \leftarrow u_i + \alpha \cdot \left(e_{ij} \cdot v_j + \sum_{t \in timestamp_i} \beta \cdot features_t \right) \tag{6}$$

$$v_i \leftarrow v_i + \alpha \cdot \left(e_{ij} \cdot u_j + \sum_{t \in timestamp_j} \gamma \cdot features_t \right)$$

Where $e_{ij} = r_{ij} - u_i^T v_j$ is the prediction error for the interaction between user i and item j, α is the learning rate, β and γ control the impact of dynamic features, and $timestamps_i$ and $timestamps_j$ denote the timestamps associated with user i's interactions and item j's interactions, respectively.

DMF-CF presents challenges in modeling temporal patterns, feature integration, and hyperparameter tuning. The model needs to effectively capture evolving preferences while avoiding overfitting and maintaining computational efficiency. To address these challenges, advanced optimization techniques, adaptive learning rate schedules, and careful selection of dynamic features should be considered. Proper cross-validation and evaluation against benchmark datasets are essential to ensure the model's effectiveness in dynamically adapting to user preferences over time.

4. Experiments

4.1. Experimental setup

To evaluate our proposed method, we deployed experiments on two MovieLens datasets. The MovieLens datasets, MovieLens-100K and MovieLens-1M (Harper & Konstan, 2016), are widely recognized and utilized benchmarks in the field of recommendation systems. These datasets were compiled by the GroupLens Research team and consist of user-item interaction data collected from the MovieLens website. MovieLens-100K contains 100,000 ratings provided by approximately 943 users for around 1,682 movies. On the other hand, MovieLens-1M is a larger dataset, containing 1 million ratings contributed by around 6,040 users for about 3,706 movies. Both datasets include additional information, such as movie genres and user demographic data, making them suitable for various recommendation algorithms, including CF, MF, and content-based methods. These datasets have been instrumental in fostering research and benchmarking the performance of recommendation algorithms due to their realistic and comprehensive nature. The details of MovieLens datasets are shown in Table 1.

Table 1

Details description of datasets

	MovieLens-100K	**MovieLens-1M**
Number of users	943	6040
Number of movies	1682	3900
Rating range	1 - 5	1 - 5
Sparsity	0.9369	0.9575

To evaluate recommender systems effectively, it is essential to establish robust baselines for comparison. We designed the list of baselines for the evaluation experiments, described as follows. One such approach is CF using Non-Negative MF (NMF) (Aghdam, Analoui, & Kabiri, 2017). This technique seeks to uncover latent factors within the user-item interaction matrix, predicting missing values and generating recommendations. A comparable method, Rec-CFSVD++, integrates CF and SVD++ (Anwar et al., 2021) to enhance recommendation accuracy. By merging user and item biases with implicit feedback, this hybrid approach extends the traditional SVD method. This probabilistic approach offers a more nuanced understanding of user-item interactions, enhancing recommendation precision. The other baseline for our experiment is the MF model that uses MF techniques, SVD (Koren et al., 2009), which effectively captures underlying patterns and relationships by decomposing the user-item matrix into latent factors. These baselines collectively constitute the baseline for evaluating the effectiveness and efficiency of the proposed method, allowing for a comprehensive analysis of its performance.

To evaluate the performance of methods, we used the evaluation metrics Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) in our evaluation experiments (Aghdam et al., 2017; Anwar et al., 2021). These metrics are fundamental evaluation metrics used to quantify the accuracy of predictions in various fields, including recommendation systems. These metrics provide insights into how well a model's predictions align with the actual observed values, enabling the assessment of predictive performance. Mathematically, MAE is expressed as follows:

$$MAE = \frac{1}{n}\sum\nolimits_{i=1}^{n} |predicted_i - actual_i \,| \tag{7}$$

Where n represents the number of instances, $predicted_i$ is the predicted value, for instance, i, and $actual_i$ is the actual observed value for instance i. A lower MAE indicates better predictive accuracy, as it reflects the average absolute distance between predictions and actual values. $RMSE$ is a slightly more sensitive evaluation metric that penalizes larger errors more heavily. It is calculated by taking the square root of the average of squared differences between predicted values and actual values. The $RMSE$ metric is formulated as:

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n} (predicted_i - actual_i)^2} \tag{8}$$

Where n represents the number of instances, $predicted_i$ is the predicted value for instance i, and $actual_i$ is the actual observed value for instance i. $RMSE$ provides a measure of the magnitude of errors and is sensitive to outliers. The lower $RMSE$ indicates better predictive accuracy, as it reflects the average distance between predictions and actual values, accounting for the squared differences.

4.2. Experimental results

We start the experiment by splitting the dataset into 70% for the training set and 30% for the testing set. The parameters to run the model are set as η = 0.0015 (learning rate), the number of epochs equals 50, and the number of factors is in a range of [10 - 100]. The experimental results are shown in Table 2 and Table 3 regarding MAE and RMSE, respectively.

As shown in Table 2, the experimental results showcase the performance metrics of various MF-based CF algorithms, namely SVD, PMF, NMF, Rec-CFSVD++, and DMF-CF, on the MovieLens-100K and MovieLens-1M datasets. Starting with the MovieLens-100K dataset, we can observe that NMF has the highest MAE score of 0.7660, followed closely by PMF at 0.7632, SVD at 0.7447, Rec-CFSVD++ at 0.7219, and DMF-CF at 0.7084. The higher MAE scores of NMF and PMF indicate that these algorithms can not capture latent factors well and generate accurate recommendations. The relatively lower MAE score of DMF-CF suggests that its improvement aligns with the dataset's characteristics. This suggests that the relationship between user latent factors, as captured by the dynamics matrix, is necessary for enhancing recommendation performance. Rec-CFSVD++ also performs slightly lower than NMF and PMF, indicating that SVD++ enhancements contribute to recommendation quality, and it surpasses the inherent capabilities of NMF and PMF. Moving to the MovieLens-1M dataset, a similar trend is observed. NMF again leads with a score of 0.7267, followed by PMF at 0.6974, SVD at 0.6964, Rec-CFSVD++ at 0.6811, and DMF-CF at 0.6683. These results align with the patterns seen in the MovieLens-100K dataset, suggesting a consistent performance order across different dataset scales.

Table 2

Comparison of the proposed method and baselines in terms of MAE metric

	SVD	**PMF**	**NMF**	**Rec-CFSVD++**	**DMF-CF**
MovieLens-100K	0.7447	0.7632	0.7660	0.7219	**0.7084**
MovieLens-1M	0.6964	0.6974	0.7267	0.6811	**0.6683**

The RMSE results, as shown in Table 3, provide a clear ranking of the algorithms in terms of prediction accuracy for both datasets. DMF-CF consistently outperforms the other algorithms in terms of RMSE, followed closely by Rec-CFSVD++, SVD, and PMF, which have higher RMSE values, while NMF falls in the middle range. The relative performance order of the algorithms is consistent across both MovieLens datasets, indicating that the algorithms' behavior remains stable when applied to different data subsets. DMF-CF consistently achieves the lowest RMSE across both datasets, suggesting that its incorporation of dynamic features enables it to capture evolving user preferences effectively, resulting in more accurate predictions. While DMF-CF performs well, the slight differences in RMSE among algorithms indicate that there might still be room for further optimization and enhancement in the model designs.

Table 3

Comparison of the proposed method and baselines in terms of RMSE metric

	SVD	**PMF**	**NMF**	**Rec-CFSVD++**	**DMF-CF**
MovieLens-100K	0.9432	0.9667	0.9748	0.9201	**0.9119**
MovieLens-1M	0.8860	0.8855	0.9201	0.8716	**0.8528**

Overall, these experimental results highlight the trade-offs among the different algorithms in terms of recommendation accuracy. While NMF and PMF seem to consistently have high errors, the DMF-CF and Rec-CFSVD++ exhibit lower MAE and RMSE scores. As shown in Figure 2, the DMF-CF outperforms in comparison with the other baselines regarding both metrics, MAE and RMSE. Additionally, the DMF-CF has unique features or aspects that differentiate it from the others, warranting further investigation into the reasoning behind its performance. DMF-CF consistently emerges as a strong performer, showcasing its potential for capturing evolving preferences. These results can guide algorithm selection, parameter tuning, and further research efforts aimed at improving recommendation accuracy and personalization capabilities.

Figure 2. The comparison between the proposed method DMF-CF and baselines regarding MAE, and RMSE metrics

5. Conclusions

The proposed DMF-CF framework leverages temporal dynamics and dynamic features to enhance the accuracy and adaptability of recommendation systems. By integrating time-varying factors, the model can generate recommendations reflecting evolving user preferences, leading to more personalized and relevant suggestions. DMF-CF presents a powerful approach for enhancing movie recommendation systems by incorporating temporal dynamics and evolving user preferences. In this study, we evaluated the performance of various algorithms, including SVD, PMF, NMF, Rec-CFSVD++, and DMF-CF, using both Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) as evaluation metrics. The results consistently demonstrated that DMF-CF outperformed the other algorithms, achieving the lowest RMSE values across two datasets, with competitive MAE scores as well. This suggests that the integration of dynamic features allows DMF-CF to accurately capture changing user behaviors over time, leading to more accurate and relevant movie recommendations. In future work, we plan to evaluate the proposed method in real-world recommendation systems.

References

Aciar, S. V., Aciar, G. I., Collazos, C. A., & Gonzalez, C. S. (2016). User recommender system based on knowledge, availability, and reputation from interactions in forums. *IEEE Revista Iberoamericana de Tecnologias Del Aprendizaje, 11*(1), 18-22. doi:10.1109/RITA.2016.2518441

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734-749. doi:10.1109/TKDE.2005.99

Aghdam, M. H., Analoui, M., & Kabiri, P. (2017). Collaborative filtering using non-negative matrix factorisation. *Journal of Information Science, 43*(4), 567-579. doi:10.1177/0165551516654354

Anwar, T., Uma, V., & Srivastava, G. (2021). Rec-CFSVD++: Implementing recommendation system using collaborative filtering and Singular Value Decomposition (SVD)++. *International Journal of Information Technology & Decision Making, 20*(4), 1075-1093. doi:10.1142/S0219622021500310

Ding, Y., & Li, X. (2005). Time weight collaborative filtering. In *Proceedings of the 14th ACM international conference on Information and knowledge management* (pp. 485-492). New York, NY: ACM.

Harper, F. M., & Konstan, J. A. (2016). The movielens datasets. *ACM Transactions on Interactive Intelligent Systems, 5*(4), 1-19. doi:10.1145/2827872

He, X., Zhang, H., Kan, M.-Y., & Chua, T.-S. (2016). Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval* (pp. 549-558). New York, NY: ACM.

Karahodza, B., Supic, H., & Donko, D. (2014). An approach to design of time-aware recommender system based on changes in group user's preferences. In *Proceedings of 2014 X International Symposium on Telecommunications (BIHTEL)* (pp. 1-4). Piscataway, NJ: IEEE.

Koren, Y. (2008). Factorization meets the neighborhood. In *Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 426-434). New York, NY: ACM.

Koren, Y. (2009). Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 447-456). New York, NY: ACM.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer, 42*(8), 30-37. doi:10.1109/MC.2009.263

Lathia, N., Hailes, S., & Capra, L. (2009). Temporal collaborative filtering with adaptive neighbourhoods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval* (pp. 796-797). New York, NY: ACM.

Lee, T. Q., Park, Y., & Park, Y.-T. (2008). A time-based approach to effective recommender systems using implicit feedback. *Expert Systems with Applications, 34*(4), 3055-3062. doi:10.1016/j.eswa.2007.06.031

Lee, T. Q., Park, Y., & Park, Y.-T. (2009). An empirical study on effectiveness of temporal information as implicit ratings. *Expert Systems with Applications, 36*(2), 1315-1321. doi:10.1016/j.eswa.2007.11.047

Liu, N. N., Zhao, M., Xiang, E., & Yang, Q. (2010). Online evolutionary collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems* (pp. 95-102). New York, NY: ACM.

Meng, J., Zheng, Z., Tao, G., & Liu, X. (2016). User-specific rating prediction for mobile applications via weight-based matrix factorization. In *Proceedings of 2016 IEEE International Conference on Web Services (ICWS)* (pp. 728-731). Piscataway, NJ: IEEE.

Nguyen, V. L., & Jung, J. J. (2020). Crowdsourcing platform for collecting cognitive feedbacks from users: A case study on movie recommender system. In *Springer series in reliability engineering book series* (pp. 139-150). Singapore: Springer.

Nguyen, V. L., Hong, M.-S., Jung, J. J., & Sohn, B.-S. (2020). Cognitive similarity-based collaborative filtering recommendation system. *Applied Sciences, 10*(12), Article 4183. doi:10.3390/app10124183

Nguyen, V. L., Nguyen, H. T., Jung, J. J., & Camacho, D. (2023). Extending collaborative filtering recommendation using word embedding: A hybrid approach. *Concurrency and Computation: Practice and Experience, 35*(16), 38-49. doi:10.1002/cpe.6232

Nguyen, V. L., Nguyen, H. T., & Jung, J. J. (2020). Content-based collaborative filtering using word embedding. In *Proceedings of the International Conference on Research in Adaptive and Convergent Systems* (pp. 96-100). New York, NY: ACM.

Nguyen, V. L., Vo, T. Q., & Nguyen, H. T. (2023). Adaptive KNN-based extended collaborative filtering recommendation services. *Big Data and Cognitive Computing, 7*(2), Article 106. doi:10.3390/bdcc7020106

Raza, S., & Ding, C. (2022). News recommender system: A review of recent progress, challenges, and opportunities. *Artificial Intelligence Review, 55*(16), 749-800. doi:10.1007/s10462-021-10043-x

Ricci, F., & Nguyen, N. Q. (2007). Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intelligent Systems, 22*(3), 22-29. doi:10.1109/MIS.2007.43

Roy, D., & Dutta, M. (2022). A systematic review and research perspective on recommender systems. *Journal of Big Data*, *9*(1), Article 59. doi:10.1186/s40537-022-00592-5

Takács, G., Pilászy, I., Németh, B., & Tikk, D. (2008). Matrix factorization and neighbor based algorithms for the Netflix prize problem. In *Proceedings of the 2008 ACM conference on Recommender systems* (pp. 267-274). New York, NY: ACM.

Wang, Y., Ma, W., Zhang, M., Liu, Y., & Ma, S. (2023). A survey on the fairness of recommender systems. *ACM Transactions on Information Systems, 41*(3), 1-43. doi:10.1145/3547333